
LSF Batch User's Quick Reference

Version 3.2

Platform Computing Corporation

LSF Batch User's Quick Reference

Copyright © 1994-1998 Platform Computing Corporation
All rights reserved.

This document is copyrighted. This document may not, in whole or part, be copied, duplicated, reproduced, translated, electronically stored, or reduced to machine readable form without prior written consent from Platform Computing Corporation.

Although the material contained herein has been carefully reviewed, Platform Computing Corporation does not warrant it to be free of errors or omissions. Platform Computing Corporation reserves the right to make corrections, updates, revisions or changes to the information contained herein.

UNLESS PROVIDED OTHERWISE IN WRITING BY PLATFORM COMPUTING CORPORATION, THE PROGRAM DESCRIBED HEREIN IS PROVIDED AS IS WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL PLATFORM BE LIABLE TO ANYONE FOR SPECIAL, COLLATERAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING ANY LOST PROFITS OR LOST SAVINGS, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS PROGRAM.

LSF Base, LSF Batch, LSF JobScheduler, LSF MultiCluster, LSF Make, LSF Analyzer, LSF Parallel, Platform Computing, and the Platform Computing and LSF logos are trademarks of Platform Computing Corporation.

Other products or services mentioned in this document are identified by the trademarks or service marks of their respective companies or organizations.

Printed in Canada

Contents

Preface	vii
Audience	vii
Common Options	vii
Related Documents	vii
Online Documentation	vii
Technical Assistance	viii
Command Reference	1
bacct	1
bbot	2
bchkpnt	2
bhist	3
bhosts	4
bhpart	5
bjobs	5
bkill	6
bmgroup	6
bmig	7
bmod	7
bparams	8
bpeek	8
bqueues	8
brestart	9
bresume	9
bstop	10
bsub	10

bswitch.....	13
btop	14
bugroup.....	14
busers.....	14
ch	14
ckpt_ld.....	15
ckpt_ld_f.....	15
lsacct.....	15
lsacctmrg.....	16
lselectible.....	16
lsgrun.....	17
lshosts	17
lsid	18
lsinfo.....	18
lsload	18
lsloadadj	19
lslogin.....	19
lsltasks.....	20
lsmake	20
lsmon	21
lsplace	21
lsrtasks.....	22
lsrun	22
lstcsh.....	23
xanalyzer.....	23
xbmod	23
xbsub	23
xlsadmin	23
xlsbatch	23
xlsmon	23
xlsf	23
Resources	24
Dynamic Resources	24
Static Resources	24

Specifying Resource Requirements.....	24
select[selectstring]	24
order[orderstring]	25
usage[usagestring].....	25
span[spanstring].....	26



Preface

Audience

This guide provides command reference information for users of LSF Base, LSF Batch, and LSF MultiCluster. Users should be familiar with executing commands in a UNIX or Windows NT environment. For detailed information see the *LSF Batch User's Guide* and the *LSF Batch Administrator's Guide*.

Common Options

All commands take the following options. They will not be shown unless they differ for a specific command:

- h Print command usage to standard error and exit.
- v Print LSF version to standard error and exit.

Related Documents

The following guides are available from Platform Computing Corporation:

- LSF Installation Guide*
- LSF Batch Administrator's Guide*
- LSF Batch Administrator's Quick Reference*
- LSF Batch User's Guide*
- LSF Batch User's Quick Reference*
- LSF JobScheduler Administrator's Guide*
- LSF JobScheduler User's Guide*
- LSF Analyzer User's Guide*
- LSF Parallel User's Guide*
- LSF Programmer's Guide*

Online Documentation

- Man pages (accessed with the `man` command) for all commands

Preface

- Online help available through the Help menu for the `xlsbatch`, `xbmod`, `xbsub`, `xbalarms`, `xbcal` and `xlsadmin` applications.

Technical Assistance

If you need any technical assistance with LSF, please contact your reseller or Platform Computing's Technical Support Department at the following address:

LSF Technical Support
Platform Computing Corporation
3760 14th Avenue
Markham, Ontario
Canada L3R 3T7

Tel: +1 905 948 8448
Toll-free: 1-87PLATFORM (1-877-528-3676)
Fax: +1 905 948 9975
Electronic mail: *support@platform.com*

Please include the full name of your company.

You may find the answers you need from Platform Computing Corporation's home page on the World Wide Web. Point your browser to *www.platform.com*.

If you have any comments about this document, please send them to the attention of LSF Documentation at the address above, or send email to *doc@platform.com*.

Command Reference

bacct

Report accounting statistics on completed batch jobs in the LSF system

```
bacct [-h] [-V] [-b] [-l] [-w] [-d] [-e] [-f logfile] [-N host_spec]
[-C time0, time1] [-S time0, time1] [-D time0, time1] [-q queuelist]
[-m hostlist] [-u userlist | -u all] [-P projectlist] [jobId ...]
```

- b Display in brief format.
- l Display in long format
- d Consider only successfully completed jobs (i.e., DONE status).
- w Display in wide format.
- e Consider only exited jobs (i.e., EXIT status).
- f logfile
 Specify the path and file name of the job log file to be analyzed.
- N host_spec
 Convert the displayed CPU time to a normalized CPU time by multiplying the CPU time by host_spec.
- C time0, time1
 Consider only those jobs whose completion or exit times were within the time interval time0 and time1.
- S time0, time1
 Consider only those jobs whose submission times were within the time interval time0 and time1.
- D time0, time1
 Consider only those jobs whose dispatch times were within the time interval time0 and time1.
- q queuelist
 Consider only those jobs submitted to the queues whose names are given in queuelist. More than one queue name must be separated by spaces and enclosed by (") or (').
- m hostlist
 Consider only those jobs executed on the hosts whose names are given in hostlist. More than one host name must be separated by spaces and enclosed by (") or (').

bbot

`-u userList | -u all`

Consider only those jobs submitted by the users whose login names and user IDs are given in `userlist`, or by `all` users. More than one user names and IDs must be separated by spaces and enclosed by (") or (').

`-P projectlist`

Consider only those jobs submitted to `projectlist`. More than one project name must be separated by spaces and enclosed by (") or (').

`jobId ...`

Consider only those jobs specified by `jobId ...`. This option overrides all other options except `-h`, `-V`, `-b`, `-l`, and `-f`.

bbot

Move a pending job to the bottom (end) of its queue. This only affects the user's own jobs.

`bbot jobId | "jobId[indexList]" [position]`

`jobId | "jobId[indexList]"`

Move the specified job or job array element.

`position`

A positive number that indicates the target position of the job relative to the end of the queue.

bchkpnt

Checkpoint one or more unfinished (running or suspended) jobs. The job must have been submitted with `bsub -k`.

`bchkpnt [-f] [-k] [-q queue] [-m host] [-u user | all] [-J jobname] [-p period] jobId | "jobId[indexList]" ...`

`-f` Checkpoint a job even if non-checkpointable (operating system-specific) conditions exist.

`-k` The job is checkpointed and killed atomically by the system.

`-q queue`

Checkpoint the jobs in the specified queue. Ignored if a job ID other than 0 is specified.

`-m host`

Checkpoint the jobs dispatched to the named by host or host group. Ignored if a job ID other than 0 is specified.

`-u user | all`

Checkpoint the jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.

- J jobname
Checkpoint the named jobs. Ignored if a job ID other than 0 is specified.
- p period
Checkpoint the job and change its checkpoint period in minutes. A period of 0 disables periodic checkpointing.
- jobId | "jobId[indexList]" ...
Checkpoint the specified jobs or job array elements.

bhist

Display the history of one or more jobs in the LSF Batch system.

JOB-BASED: `bhist [-b] [-w] [-l] [-a] [-d] [-p] [-s] [-t]`
`[-f logfile | -n num_logfiles] [-C time0,time1] [-S time0,time1]`
`[-D time0,time1] [-q queue] [-m host] [-u user | all] [-J job]`
`[-P project] [-N spec] [jobId | "jobId[indexList]" ...]`

CHRONOLOGY-BASED: `bhist [-t] [-f logfile] [-T time0,time1]`

- b Display the job history in a brief format. Default: display summary format.
- w Display the job history in a wide format without truncation. Default: display summary format.
- l Display the job history in a (long) multi-line format, giving detailed information about each job. Default: display summary format.
- a Display all, both finished and unfinished, jobs. Default: finished jobs are not displayed.
- d Display only the finished jobs.
- p Display only the pending jobs.
- s Display only the suspended jobs. If option -l or -b is also specified, show the reason why each job was suspended.
- t Display job events chronologically. Default: display events occurring in the past week.
- f logfile
Specify the file name of the event log file. Either an absolute or a relative path name may be specified. Default: use the system event log file `lsb.events`.
- n num_logfiles
Specify the number of event log files that are searched. A value of 0 causes all log files in the cluster's log directory to be searched. Default: 1.
- C time0,time1
Display only those jobs whose completion or exit times were between time0 and time1. Default: display all jobs that have completed or exited.

bhosts

-S time0,time1
Display only those jobs whose submission times were between time0 and time1.
Default: display all jobs that have been submitted.

-D time0,time1
Display only those jobs whose dispatch times were between time0 and time1.
Default: display all jobs that have been dispatched.

-q queue
Display jobs submitted to the specified queue only. Default: display all queues.

-m host
Display jobs dispatched to the specified host only. Default: display all hosts.

-u user | all
Display jobs submitted by the named user or by all users. Default: display the jobs submitted by the user who invoked this command.

-J job
Display all jobs with the specified name.

-P project
Display only jobs belonging to the named project. Default: display jobs belonging to all projects.

-N spec
Display CPU time consumed by the job. The appropriate CPU scaling factor for the specified host, or defined host model, is used to normalize the actual CPU time consumed by the job.

jobId | "jobId[indexList]" ...
Display the specified jobs.

-T time0,time1
Consider only those job events occurring between time0 and time1.

bhosts

Display information about the server hosts in the LSF Batch system.

HOST-BASED: `bhosts [-R res_req] [-w | -l] [host ... | cluster]`

RESOURCE-BASED: `bhosts [-s] resource`

-R res_req
Display information about the hosts that meet the specified resource requirements.
Default: display information for all hosts in the local cluster.

-w
Display fields in a wide format without truncation.

-l
Display information in a (long) multi-line format.

host ... | cluster
Display information on one or more hosts or cluster. Default: display information for all hosts in the local cluster.

-s Display numeric shared resource information for the host on a per-resource basis.
resource
 Specifies one or more numeric shared resources about which information will be displayed. Default: all numeric shared resources.

bhpart

Display information about host partitions in the LSF Batch system.

bhpart [-r] [host_partition ...]

-r Recursively display the entire share information tree associated with the host partition.
host_partition ...
 Display information about the named host partition(s). Default: all.

bjobs

Display the status and other information about jobs in the LSF Batch system.

bjobs [-w | -l] [-a] [-d] [-p] [-s] [-r] [-A] [-q queue]
 [-m host | cluster] [-u user | all] [-J jobname] [-P project] [-N spec]
 [jobId | "jobId[indexList]" ...]

-w Display fields in a wide format without truncation.
-l Display information in a (long) multi-line format.
-a Display information about all jobs, including unfinished jobs (pending, running or suspended) and recently finished jobs.
-d Display only recently finished jobs.
-p Display only pending jobs, with the reasons they were not dispatched during the last dispatch turn.
-s Display only suspended jobs, showing the reason for suspension.
-r Display only running jobs.
-A Display summarized information about job arrays.
-q queue
 Display only jobs in the named queue.
-m host | cluster
 Display only jobs dispatched to the named host or host group.
-u user | all
 Display jobs submitted by the named user or by all users. Default: display the jobs submitted by the user who invoked this command.
-J jobname
 Display all jobs with the specified name.

bkill

- P project
Display only jobs belonging to the named project. Default: display jobs belonging to all projects.
- N spec
Display CPU time consumed by the job. The appropriate CPU scaling factor for the specified host, or defined host model, is used to normalize the actual CPU time consumed by the job.
- jobId | "jobId[indexList]" ...
Display the specified jobs.

bkill

Send a signal to one or more unfinished batch jobs.

```
bkill [-l] [-s value | name] [-q queue] [-m host] [-u user | all]  
[-J jobname] [jobId | "jobId[indexList]" ...]
```

- l
Display the set of signal names supported. (This is a subset of those supported by /bin/kill and is platform dependent).
- s value | name
Send the signal specified by value or name to the specified jobs. Default: SIGKILL or 9.
- q queue
Send a signal to the jobs in the specified queue.
- m host
Send a signal to the jobs dispatched to the specified host or host group. Ignored if a job ID other than 0 is specified.
- u user | all
Send a signal to the jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.
- J jobname
Send a signal to the named jobs. Ignored if a job ID other than 0 is specified.
- jobId | "jobId[indexList]" ...
Send a signal to the specified jobs or job array elements.

bmgroup

Display LSF Batch host group membership. If a group member is a group itself (subgroup), then a slash '/' is appended to its name.

```
bmgroup [-r] [group ...]
```

- r
Expand the group membership recursively.
- l
Display information in a (long) multi-line format.

group ...

Display the membership of the named group(s). Default: display the members of all groups.

bmig

Migrate one or more unfinished (running or suspended) jobs to another host. The job must have been submitted using the *bsub* options *-r* or *-k*.

```
bmig [-f] [-m host ...] [-u user | all] [-J jobname]
[jobId | "jobId[indexList]" ...]
```

-f Force the job to be migrated and restarted even if non-restartable conditions exist. Non-restartable conditions are specific to the operating system.

-m host ... Restrict candidate jobs for migration to those from the named host(s).

-u user | all Migrate the jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.

-J jobname Migrate the named jobs. Ignored if a job ID other than 0 is specified.

jobId | "jobId[indexList]" ... Migrate the specified jobs or job array elements.

bmod

Modify the options of a previously submitted job. *bmod* uses a subset of the *bsub* options. To reset an option to its default value, use the option string followed by *'n'*.

```
bmod [-x | -xn] [-r | -rn] [-N | -Nn] [-B | -Bn] [-q queue ... | -qn]
[-m host ... | -mn] [-n minimum[,maximum] | -nn] [-R res_req | -Rn]
[-J jobname | -Jn] [-b begin | -bn] [-t terminate | -tn]
[-i infile | -in] [-o outfile | -on] [-e errfile | -en]
[-u mail_user | un] [[-f "lfile op [rfile)"] ... | -fn] ]
[-c cpu[/host] | -cn] [-W run [/host ] | -Wn] [-F file | -Fn]
[-M mem | -Mn] [-D data | -Dn] [-S stack | -Sn] [-C core | -Cn]
[-k dir [period] | -kn] [-L shell | -Ln] [-G user_group | Gn]
[-P project | -Pn] [-w depend | -wn] [-E "pre_exec [arg ...]" | -En]
[-Z "new_command"] jobId | "jobId[indexList]"
```

jobId Modify the options of the specified job.

"jobId[indexList]" Modify the options of the specified job array elements.

bparams

bparams

Display information about the configurable LSF Batch system parameters. By default only display the most ‘interesting’ parameters.

```
bparams [-l]
```

- l Display all the parameters along with brief descriptions and their corresponding names.

bpeek

Display the standard output and standard error that has been produced by an unfinished batch job in the LSF Batch system.

```
bpeek [-f] [-q queue | -m host | -J jobname | jobId | "jobId[indexList]"]
```

- f Display the output of the job using the command “tail -f”. Default: use the command “cat”.
- q queue Display the output of the most recently submitted job in the specified queue.
- m host Display the output of the most recently submitted job that has been dispatched to the specified host.
- J jobname | jobId | "jobId[indexList]" Display the output of the most recently submitted job that has the specified job name, the output of the job with the given job ID, or the output of the specified job array element. Default: display the output of the most recently submitted job that satisfies options -q, -m, or -J.

bqueues

Display information about batch queues in the LSF Batch system.

```
bqueues [-w | -l | -r] [-m host | cluster] [-u user] [queue ...]
```

- w Display fields in a wide format without truncation.
- l Display information in a (long) multi-line format.
- r Same as -l, but also display recursively the share account tree of the fairshare queue (if it is defined).
- m host | cluster Display queues from which jobs may be dispatched to the named host or host group. Default: all hosts. Otherwise, display the queues in the named remote cluster.

`-u user`
Display the queues to which the named user or user group may submit batch jobs.
Default: all users.

`queue ...`
Display information about the named queue(s).

brestart

Submit a job to be restarted from the checkpoint files in the specified directory. `brestart` uses a subset of the `bsub` options.

```
brestart [-x] [-N] [-B] [-f] [-q queue] [-m host ...] [-b begin]
[-t terminate] [-w depend] [-E "pre_exec [arg ...]" ] [-c cpu[/spec]]
[-W run[/spec]] [-F file] [-D data] [-S stack] [-C core] [-M mem]
[-G user_group] dir [jobId]
```

`-f` Force the job to be restarted even if non-restartable conditions exist. Non-restartable conditions are operating system-specific.

`dir` The checkpoint directory specified with `bsub -k`.

`jobId` The last job ID of the checkpointed job.

bresume

Resume one or more unfinished batch jobs.

```
bresume [-q queue] [-m host] [-u user | all] [-J jobname]
[jobId | "jobId[indexList]" ...]
```

`-q queue`
Resume the jobs in the specified queue.

`-m host`
Resume the jobs dispatched to the specified host or host group. Ignored if a job ID other than 0 is specified.

`-u user | all`
Resume the jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.

`-J jobname`
Resume the named jobs. Ignored if a job ID other than 0 is specified.

`jobId | "jobId[indexList]" ...`
Resume the specified jobs or job array elements.

bstop

bstop

Suspend one or more unfinished batch jobs.

```
bstop [-q queue] [-m host] [-u user | all] [-J jobname]
[jobId | "jobId[indexList]" ...]
```

-q queue

Stop the jobs in the specified queue.

-m host

Stop the jobs dispatched to the specified host or host group. Ignored if a job ID other than 0 is specified.

-u user | all

Stop the jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.

-J jobname

Stop the named jobs. Ignored if a job ID other than 0 is specified.

jobId | "jobId[indexList]" ...

Stop the specified jobs or job array elements.

bsub

Submit a job for execution in the LSF Batch system. If no command is given, bsub reads job commands from the standard input. If the standard input is a controlling terminal, bsub prompts for commands.

```
bsub [-H] [-x] [-r] [-N] [-B] [-I | -Ip | -Is | -K] [-q queue ...]
[-m host[+[level]] ...] [-n min[,max]] [-R res_req]
[-J jobname | "arrayName[indexList]" ] [-b begin] [-t terminate]
[-i infile] [-o outfile] [-e errfile] [-u user]
[[-f "lfile op [rfile]"...] [-E "pre_exec [arg ...]" ] [-c cpu[/spec]]
[-W run[/spec]] [-F file] [-M mem] [-D data] [-S stack] [-C core]
[-k "dir [period]" ] [-w depend] [-L shell] [-P project] [-G user_group]
[command [arg ...]]
```

-H Hold the job in the PSUSP state when the job is submitted.

-x Exclusive execution mode.

-r Specifies that the job can be rerun.

-N Notify the submitter by mail when the job finishes. If -o option is not given, the standard output from the job is included in the message.

-B Notify the submitter by mail when the job is dispatched and begins execution.

-I Submit an interactive job. bsub blocks until the job is terminated. See -i, -o, and -e options. Default: job interacts with user's terminal.

-Ip Submit an interactive job and create a pseudo-terminal when the job starts.

-
- I s Submit an interactive job and create a pseudo-terminal with shell mode support when the job starts. Shell mode support is required for applications which redefine the CTRL-C and CTRL-Z keys.
 - K Submit a batch job and wait for the job to complete
 - q queue ...
 Submit the job to the specified queue(s). Default: the system default queue.
 - m host[+[level]] ...
 Limit candidate hosts for executing this job to the specified host(s). '+' is used to specify preference. level is a positive number specifying the preference level of a host.
 - n min[,max]
 The minimum and maximum number of processors requested to run a (parallel) job. Default: one processor.
 - R res_req
 Execute the task on a host that meets the resource requirements. Default: type==local unless a model or boolean resource is specified, in which case type==any order[rlm:pg].
 - J jobname | "arrayName[indexList]"
 jobname assigns the specified character string to the job.
 arrayName[indexList] submits a job array named arrayName with the elements and structure indexList.
 - b begin
 Dispatch the job for execution on or after the specified time.
 - t terminate
 The job termination deadline. Default: allow the job to run as long as its resource limits permit.
 - i infile
 The job gets its standard input from the specified file. Default: /dev/null (UNIX).
 - o outfile
 Write the standard output of the job to the specified file. Default: sent by mail to the user.
 - e errfile
 Write the standard error output of the job to the specified file.
 - u user
 Send mail to a specified email address. Default: send mail to the submitting user according to the format specified in the lsf.conf file.
 - f "[lfile op [rfile]]" ...
 Copy a file between the local (submission) host and the remote (execution) host. op is one of '>', '<', '<<', or '>>' (or '<>'). op is illegal without at least one of lfile or rfile.

- E "pre_exec [arg ...]"
Execute the pre-execution command on the host to which the job is dispatched to run (or on the first host selected for a parallel job) before actually running the job. If the pre-execution command exits with 0, start the primary job; otherwise the job remains pending.
- c cpu[/spec]
Set the total CPU time limit for this job. The optional `spec` defines the appropriate host CPU scaling factor. Default: no limit.
- W run[/spec]
Set the wall-clock run time limit of this job. `spec` is the same as in the `-c` option. Default: no limit.
- F file
Set a per-process (soft) file size limit for each of the processes belonging to this job. Default: no limit.
- M mem
Set the total process resident set size limit in KB for the whole job. Default: no limit.
- D data
Set a per-process (soft) data segment size limit for each of the processes that belong to this job. Default: no soft limit.
- S stack
Set a per-process (soft) stack segment size limit for each of the processes that belong to this job. Default: no soft limit.
- C core
Set a per-process (soft) core file size limit for all the processes that belong to this job. Default: no soft limit.
- k dir [period]
This job can be checkpointed. The checkpoint directory is a relative or absolute path name, and is used for restarting the job (see *brestart*). Multiple jobs can checkpoint into the same directory. Optionally, a checkpoint period in minutes may be specified.

-w depend

Only when the dependency condition is satisfied (TRUE), will the job be considered for dispatch. The expression is composed of job or system conditions using '&&', '|', and '!' logic operators. Conditions are job status, calendar, file event, or user event.

Job status conditions: `started(jobId|jobName)`, `done(jobId|jobName)`, `exit(jobId|jobName)`, and `ended(jobId|jobName)`

Calendar conditions (calendars are only available with LSF JobScheduler):

`calendar(cal_spec)` or `cal(cal_spec)`

File event conditions:

`file(arrival(location)|exist(location)|size(location)|age(location))`

User event conditions: `event(spec)` or `ev(spec)`

-L shell

Specifies the name of the login shell used to initialize the execution environment.

-P project

Specifies the name of the project that this job will be charged to.

-G user_group

Specifies the name of the LSF user group to which the job will belong.

bswitch

Switch one or more unfinished (running or suspended) jobs from one queue to another.

```
bswitch [-q queue] [-m host] [-u user | all] [-J jobname] destination
[jobId | "jobId[indexList]" ...]
```

-q queue

Switch jobs from the specified queue. Ignored if a job ID other than 0 is specified.

-m host

Switch jobs dispatched to the named by host or host group. Ignored if a job ID other than 0 is specified.

-u user | all

Switch jobs submitted by the named user or by all users. Ignored if a job ID other than 0 is specified.

-J jobname

Switch the named jobs. Ignored if a job ID other than 0 is specified.

destination

The destination queue to which jobs will be switched.

jobId | "jobId[indexList]" ...

Switch the specified jobs or job array elements.

btop

btop

Move a pending job to the top (beginning) of its queue. This only affects the user's own jobs.

```
btop jobId | "jobId[indexList]" [position]
```

```
jobId | "jobId[indexList]"
```

Move the specified job or job array element.

```
position
```

A positive number that indicates the target position of the job relative to the beginning (for `btop`) of the queue.

bugroup

Display LSF Batch user membership. If a group member is a group itself (subgroup), then a slash '/' is appended to its name.

```
bugroup [-r] [-l] [group ...]
```

```
bmgroup [-r] [group ...]
```

```
-r      Expand the group membership recursively.
```

```
-l      Display information in a (long) multi-line format.
```

```
group ...
```

Display the membership of the named group(s). Default: display the members of all groups.

busers

Display information about LSF Batch users and user groups.

```
busers [user ... | all]
```

```
user ... | all
```

Display information about the named user(s) or user group(s) or about all users and user groups. Default: display information about the invoker.

ch

Change to a host with the same execution environment on which subsequent commands will be executed.

```
ch [-t] [-S] [host]
```

```
-t      Timing option. Display amount of time each subsequent command takes to execute.
```

-S Start the remote tasks with shell mode support. Shell mode support is required for applications which redefine the CTRL-C and CTRL-Z keys. Default: shell mode support is disabled.

hostname

Change host to **hostname**. Default: local host.

Built-in commands:

cd [dir]

Change current working directory to **dir**. Default: user's HOME directory.

ch [hostname]

Change the current host to **hostname**. Default: host where **ch** command was first executed.

post [command [arg ...]]

Execute **command** in the background on the current host. Display a unique **taskID**. Output from background jobs may be sent to your monitor which could cause display problems. Default: display all currently running background tasks.

contact taskID

Put the previously posted background task into the foreground. **taskID** was returned by the **post** command.

exit Exit **ch** if there are no posted commands running.

ckpt_ld

Replacement linker for C programs to be checkpointed.

ckpt_ld [options ...] file

ckpt_ld_f

Replacement linker for Fortran programs to be checkpointed.

ckpt_ld_f [options ...] file

lsacct

Report accounting statistics on completed RES tasks in the LSF system.

lsacct [-h] [-V] [-l] [-f logfile] [-u userlist | -u all] [-m hostlist] [-C time0, time1] [-S time0, time1] [pid ...]

-l Display individual statistics about each selected job.

-f logfile

Specify the file name of the task log file to be analyzed.

lsacctmrg

`-u userlist | -u all`
Consider only those tasks belonging to the users whose login names and user IDs are given in `userlist`, or by all users if the reserved user name `all` is specified.

`-m hostlist`
Consider only those tasks executed on the hosts whose names are given in `hostlist`.

`-C time0, time1`
Consider only those tasks that completed or exited within the time interval `time0` and `time1`.

`-S time0, time1`
Consider only those tasks that began executing within the time interval `time0` and `time1`.

`pid ...` Consider only those tasks specified by `pid ...`. This option overrides all other options except `-h`, `-V`, `-l` and `-f`. The default is to consider all the tasks satisfying other options.

lsacctmrg

Merge all records in LSF task log files in chronological order according to their completion time.

`lsacctmrg [-h] [-V] [-f] logfile... dest_logfile`

`-f` Force to override the destination file without first requesting confirmation.

`logfile ...`
Specifies the set of LSF task log files to be merged.

`dest_logfile`
Specifies the destination file into which the task log files are to be merged. This file must not be one named in `logfile`.

lseligible

Display the remote execution eligibility of a task.

`lseligible [-h] [-V] [-q] [-r] [-s] task_name`

`-r` Remote mode of operation.

`-q` Quiet mode.

`-s` Silent mode.

lsgrun

Execute a task on the specified group of hosts.

```
lsgrun [-i] [-S] [-P] [-p] [-v] -f file | -m host ... | -n needed
[-R res_req] [task[arg... ]]
```

- i Interactive operation mode. The user is asked if the task will be executed on all hosts.
- S Create a pseudo-terminal with shell mode support when starting a remote task. Shell mode support is required for applications which redefine the **CTRL-C** and **CTRL-Z** keys.
- P Create a pseudo-terminal when starting a remote task.
- p Parallel run mode. If specified, the task is executed on all hosts simultaneously without pseudo-terminal support.
- v Verbose mode. The name of the selected host is displayed.
- f file
Specify a file that contains a list of host names. This option is exclusive of options -m, -n, and -R.
- m host ...
Specify the candidate host(s) to execute task. This option is exclusive of options -f, -n, and -R.
- n needed
Specify the number of processors to run task. A host may be used to start several tasks if it is a multiprocessor. This option can be used with option -R, and is exclusive of options -f and -m.
- R res_req
Execute the task on a host that meets the specified resource requirements. This option can be used with option -n. Default: `type==local` unless a model or Boolean resource is specified, in which case `type==any order[r15s:pg]`.
- task[arg ...]
The name of the task to be executed. If it is not specified, then the user is prompted to enter a command terminated by **CTRL-D** or **EOF**.

lshosts

Display information about LSF host configuration, including: host name, host type, host model, CPU normalization factor, number of CPUs, total memory, and available resources, on either a per-host or per-resource basis.

```
HOST-BASED: lshosts [-w | -l] [-R res_req] [host ... | cluster ...]
```

```
RESOURCE-BASED: lshosts -s [resource ...]
```

- w Display information in a wide format.

lsid

-l Display information in a (long) multi-line format.

-R res_req Display information about the hosts that meet the specified resource requirements.
Default: display information for all hosts.

host ... | cluster ... Display configuration information for the named host(s) or cluster(s). Default:
display information for all hosts.

-s Display static shared resource information on a per-host basis.

resource ... Specifies the static shared resource(s) about which information will be displayed.
Default: all static shared resources.

lsid

Display the LSF release version, the name of the local load sharing cluster, and the name of its master LIM host.

lsinfo

Display information about the LSF configuration, including available resources, host types and host models.

lsinfo [-l] [-r] [-m] [-t] [resource ...]

-l Display resource information in a (long) multi-line format.

-r Display information about configured resources.

-m Display information about configured host models.

-t Display information about configured host types.

resource ... Display information about the named resource(s). Default: all resources.

lsload

Display load information on hosts, in order of increasing load, on either a per-host or per-resource basis.

HOST-BASED: lsload [-N | -E] [-l] [-R res_req] [-I index] [-n number]
[host ... | cluster ...]

RESOURCE-BASED: lsload -s [resource ...]

-N Return normalized CPU run queue length load indices. CPU run queue values are normalized using the number of CPUs and the CPU factors of the hosts. Default: return raw load indices. Options -N and -E are mutually exclusive.

- E Return effective CPU run queue length load indices. CPU run queue values are normalized using the number of CPUs of the hosts. Default: return raw load indices. Options -N and -E are mutually exclusive.
- l Display information in a long format.
- R res_req
 Display information about the hosts that meet the specified resource requirements. Default: type==any order[r1m:pg].
- I index
 Display only the specified load indices. List of load index names are separated by ':' (r1m:pg:ut).
- n number
 Display up to the specified number of hosts that best satisfy the resource requirements. Default: display all hosts that satisfy the resource requirements.
- host ... | cluster ...
 Display information for the named host(s) or cluster(s). Default: display information for all hosts. If -R is specified, display all hosts that satisfy the resource requirements.
- s Display the load on a per-resource basis.

lsloadadj

Adjust the load condition data of hosts by sending a load adjustment request to the LIM.

```
lsloadadj [-h] [-V] [-R res_req] [host_name[:num_task] ...]
```

- R res_req
 Specify the resource requirements of the exported tasks.
- host_name[:num_task] ...
 Specify a list of hosts for which the load is to be adjusted. num_task indicates the number of tasks to be started on the host. The default value is one. If no host name is given, the load of the local host is adjusted.

lslogin

Select a lightly loaded host that has few users and log in remotely, using the rlogin command. lslogin accepts all the options used by rlogin, including the host option. It has additional options to support its LSF features.

```
lslogin [-m host | cluster] [-v] [-R res_req] [rlogin_options]
```

- m host | cluster
 Specify the candidate host to which to log in. When a cluster is specified, the most suitable host in that cluster is selected.
- v Display the name of the selected host.

lsftasks

`-R res_req`
Select a host satisfying the specified resource requirements. Default: `type==any`
`order[ls:rlm]`.

lsltasks

Display or update the user's local task lists. These task lists are maintained by LSF on a per user basis to store the resource requirements of tasks. Default: display the task list in multi-column format.

`lsltasks [+ | - task[/res_req] ...]`

`+ task[/resreq] ...`
Add the task(s) to the task list, along with any optional resource requirements.

`- task ...`
Remove the task(s) from the task list.

lsmake

A modified version of GNU Make which executes make tasks in parallel on LSF hosts. `lsmake` accepts all the options used by GNU Make (*makeoptions*). It also has additional options to support its LSF features.

`lsmake [-m host ... | [[-R res_req] [-j n]]] [-F res_req] [-P period]
[-c k] [-M] [-V] [-E] [makeoptions] [target ...]`

`-m host ...`
Specify the candidate host(s) to execute task. This option cannot be used if `-j` and/or `-R` is specified.

`-R res_req`
Use hosts satisfying the specified resource requirements. Used along with the `-j` option. Default: use hosts that are the same type and are lightly loaded in CPU and memory.

`-j n` Select at most *n* CPU slots. A CPU slot corresponds to a host, with multiprocessors having several CPU slots. This option cannot be used if `-m` is specified.

`-F res_req`
File server saturation threshold requirement. If the file server becomes overloaded, the parallelism is gradually reduced until the load is within the threshold. Default: do not consider file server load.

`-P period`
CPU slot reselection period in minutes. This ensures that lightly loaded hosts are used. Default: do not reselect slots.

`-c k` Start *k* tasks concurrently in each CPU slot. Default: only one task is started in each CPU slot.

- M Process submakes in parallel. Default: process submakes sequentially.
- V Turn on verbose remote execution mode.
- E Set environment variables for each task executed remotely. Default: when `lsmake` first starts up, set the environment variables once on the remote hosts.

lsmon

Full screen LSF monitoring utility displaying dynamic load information of hosts. `lsmon` supports all the *lsload* options, plus the additional `-i` and `-L` options. It also has run-time options.

```
lsmon [-N | -E] [-n number] [-R res_req] [-I index] [-i interval]
[-L logfile] [host ...]
```

- i interval
Update interval of load information. Default: 10 seconds.
- L logfile
Save the load information into the specified file.

Run-time options:

- ^L Refresh the screen.
- i Input a new update interval.
- n Input a new number of hosts to display.
- N Switch between displaying normalized and raw CPU load indices.
- E Switch between displaying effective and raw CPU load indices.
- R Input new resource requirements.
- q Quit `lsmon`.

lsplace

Query LIM daemon for a placement decision. This command is normally used in an argument to select hosts for other commands.

```
lsplace [-L] [-R res_req] [-n needed] [-w wanted] [host ...]
```

- L Attempt to place tasks on as few hosts as possible.
- R res_req
Only the hosts that meet the specified resource requirements are considered. Those with lowest load are selected. Default: select for CPU and memory intensive tasks, needing the same type of hosts as the local host.
- n needed
The minimum number of CPUs required. Default: 1.

lsrtasks

- w wanted
The maximum number of CPUs requested. Default: 1.
- host ...
Use only the specified hosts as candidates for the host selection.
- Command exit status:
 - 0 Normal.
 - 1 Not enough hosts available.
 - 10 Failure detected in the LSF system.
 - 1 Other errors.

lsrtasks

Display or update the user's remote task lists. These task lists are maintained by LSF on a per user basis to store the resource requirements of tasks. Default: display the task list in multi-column format.

- ```
lsrtasks [+ | - task[/res_req] ...]
```
- + task[/resreq] ...  
Add the task(s) to the task list, along with any optional resource requirements.
  - task ...  
Remove the task(s) from the task list.

## lsrun

Submit a task to the LSF system for execution, possibly on a remote host.

- ```
lsrun [-S] [-P] [-l] [-v] [-m host ... | cluster ...] [-R res_req]  
task [arg ...]
```
- S Create a pseudo-terminal with shell mode support when starting a remote task. Shell mode support is required for applications which redefine the **CTRL-C** and **CTRL-Z** keys.
 - P Create a pseudo-terminal when starting a remote task.
 - l Always execute the task locally if remote execution fails. Default: execute locally only if the local host also satisfies the resource requirements.
 - v Verbose mode. The name of the selected host is displayed.
 - m host ... | cluster ...
Specify the candidate host(s) or cluster(s) for executing the task. If none of the candidate hosts satisfy the resource requirements, the task will not run. If only one host is specified, this option overrides the **-R** option and the task will be executed on that host.

`-R res_req`

Execute the task on a host that meets the resource requirements. If `-R` and `-m` are not specified, the resource requirements from the remote task list are used. Default: `type==local` unless a model or boolean resource is specified, in which case `type==any` `order[rlm:pg]`.

lstcsh

A load sharing version of `tcsh` (an enhanced version of the C shell). `lstcsh` accepts all the options used by `tcsh`, plus the additional `-L` option.

`lstcsh [-bcdefimnstvVxXL] [arg ...]`

`-L` Execute an `lstcsh` script with load sharing enabled.

xanalyzer

A graphical user interface to the LSF Analyzer application used for LSF Batch system workload analysis.

xbmod

A graphical user interface for modifying the options of a job submitted to the LSF Batch system.

xbsub

A graphical user interface for submitting a job to the LSF Batch system.

xlsadmin

A graphical user interface for controlling and modifying the LSF Base and Batch system and their configurations.

xlsbatch

A graphical user interface for monitoring jobs and other aspects of the LSF Batch system.

xlsmon

A graphical user interface for monitoring the load on the hosts in an LSF cluster.

xlsf

A graphical user interface (GUI) console used to access the other LSF Batch system GUIs.

Dynamic Resources

Resources

There are two types of resources for a task: dynamic resources and static resources. Use *lsinfo* to see a complete list of dynamic and static resources available on your system.

Dynamic Resources

LSF defines several dynamic resources. Each dynamic resource corresponds to a Load Index. Load indices are used to determine how busy a host is (see *lsload*).

Static Resources

Any unchanging characteristic or attribute of a host that can be useful in selecting hosts for remote tasks may be considered a static resource. Static resource requirements of a task limit the eligible machine set for the task, or the set of machines that can execute the task.

Some static boolean resources are site-dependent, defined by the LSF administrator in the LSF configuration files.

Use *lshosts* to see what static resources are available on each host.

Specifying Resource Requirements

A resource requirement string describes the resources a job needs. LSF uses resource requirements to select hosts for remote execution and job execution.

The syntax of a resource requirement string is as follows. The brackets are real, not an element of syntactic notation.

```
select[selectstring] order[orderstring] rusage[usagestring] span[spanstring]
```

select[selectstring]

The keyword `select` and its associated brackets can be omitted.

The `selectstring` specifies the characteristics a host must have to match the resource requirement. It is a logical expression built from resource names combined with logical and arithmetic operators using standard C language syntax and precedence.

Boolean resources have a value of 1 if they are defined for a host, and 0 otherwise. Non-zero arithmetic values are treated as logical `TRUE`, and zero as logical `FALSE`.

The special values `any` and `local` (meaning the same value as the local host) may be used with the resources `type` and `model`.

Note:

The values of `r15s`, `r1m`, and `r15m` used for selection are the effective load indices returned by `lsload -E`.

For `lshosts`, `lsload`, `lsmon`, and `lslogin` the default is `type==any`. For `lsplace`, `lsrun`, `lsgrun`, `lsmake`, and `bsub` the default is `type==local` unless a model or boolean resource is specified, in which case it is `type==any`.

order[orderstring]

The `orderstring` allows the selected hosts to be sorted according to the values of resources. The syntax of `orderstring` is:

```
[ - ]res[:[ - ]res]...
```

Each `res` must be a dynamic load index (see `lsload`) or an external load index defined by the LSF administrator.

Note:

The values of `r15s`, `r1m`, and `r15m` used for sorting are the normalized load indices returned by `lsload -N`.

The ordering begins with the rightmost index in the string and proceeds from right to left. The hosts are sorted into order based on each load index, and if more hosts are available than were requested, the LIM drops the least desirable hosts according to that index. The remaining hosts are then sorted by the next index.

After the hosts are sorted by the leftmost index in `orderstr`, the final phase of sorting orders the hosts according to their status, with hosts that are currently not available for load sharing (that is, not in the `ok` state) listed at the end.

The default is `r1m:pg` (except for `lslogin` which is `ls:r1m`).

usage[usagestring]

The `usagestring` specifies the expected resource usage of the task. This is used to specify resource reservation the load and for mapping tasks onto hosts during a placement decision. The syntax of `usagestring` is:

```
res=value[:res=value]...[:res=value][:duration=value][:decay=value]
```

Here `res` can be any load index. The `value` parameter is the initial reserved amount.

The default is `r1m:r15s:mem:ut`.

span(spanstring)

span[spanstring]

The `spanstring` specifies the locality of a parallel batch job. There are two possible values for the `spanstring`.

If defined as `span[hosts=1]`, all processors allocated to this job must be on the same host. If defined as `span[ptile=N]`, up to N processor(s) on each host should be allocated to the job.